NATIONAL IMAGERY TRANSMISSION FORMAT STANDARD (NITFS)
REQUEST FOR CHANGE (RFC)

RFC CONTROL NUMBER 95-051
(To be filled in by NTB Secretary)

DATE SUBMITTED 3/1/95 DATE RECEIVED 3/2/95

ORIGINATOR Ken Whitson
TELEPHONE (703) 275-5088

ORGANIZATION TYPE User

MAILING ADDRESS Central Imagery Office
8401 Old Courthouse Road
Vienna, VA 22182-3280

PRIORITY Routine

FUNCTION Operational

DOCUMENT NUMBER MIL-STD-2500 PAGE
DOCUMENT NITFS Format Version 2.0 PARAGRAPH

PROBLEM DESCRIPTION
Addition of Integrity Seals for National Imagery Transmission Format (NITF) files. (See attachment).

RECOMMENDED WORDING

RATIONALE

REMARKS

TOTAL COST OF IMPLEMENTATION

PROPOSED TIMEFRAME OF IMPLEMENTATION

ANTICIPATED USER IMPACT

NTB REVIEW DATE NTB RECOMMENDATION
SUBSTANTIVE ISSUES

DATE SUBMITTED TO ISMC
ISMC REVIEW DATE

DATE SUBMITTED TO DISA

ISMC DECISION

IMPLEMENTATION DATE

INTEGRITY SEALS
FOR THE
NATIONAL IMAGERY TRANSMISSION FORMAT
STANDARD

1.0 STATUS

This RFC proposes the addition of integrity seals for National Imagery Transmission Format (NITF) [1] files that bind headers and subheaders to the object or data the headers and subheaders represent. These integrity seals will allow trusted software and equipment to verify that a NITF header and subheader and associated object (e.g. image, symbol, text, label, data extensions, and reserved segments) have not changed since the seal was applied.

2.0 INTRODUCTION

An NITF file is composed of a header, subheaders, and objects. This level of abstraction is offered for this RFC for simplification. The header contains various data fields that represent the NITF file. The remainder of the file is composed of subheaders and objects. Subheaders are similar to the NITF file header but only represent the object that it proceeds. For the purpose of this RFC, the term objects refers to NITF images, symbols, labels, text, data extensions, and any reserved segments. Figure 1 illustrates the format of an NITF file. Some of the information contained in the NITF header and subheaders is security-relevant (e.g. the security level of the objects). This information should be strongly bound to the object that it represents so that any modification to either the header or the object will be detected. This binding is accomplished by adding an integrity seal to each object and to the entire NITF file. By verifying the integrity seal, the verifier can be assured that the NITF file headers, subheaders, and objects have not been tampered with since the integrity seal was applied. The integrity seal can be verified by the recipient's software, a trusted database, or a communication gateway/guard and used to enforce a security policy.

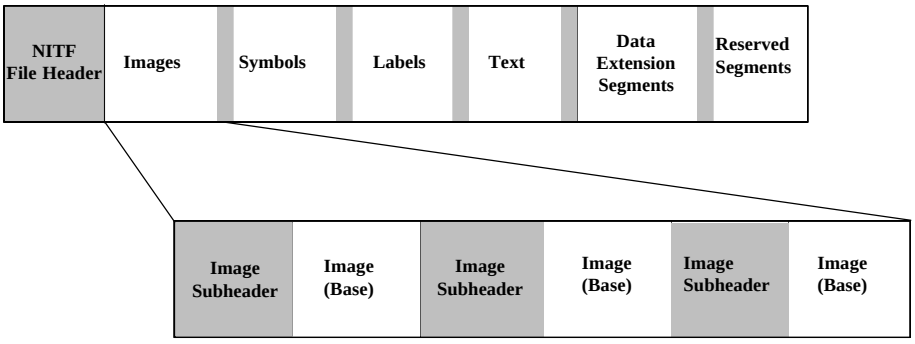

Figure 1. NITF File Structure

3.0 THE INTEGRITY SEAL

An NITF integrity seal is created by adding a digital signature that was formed from the concatenation of the NITF file header and the rest of the NITF file, or the concatenation of a subheader and the object it represents. This integrity seal is appended to the file or the object it represents. The entire NITF integrity seal is actually composed of a single file integrity seal and many object integrity seals. The use of more then a single integrity seal allows for the integrity of an object to be maintained when another object is modified. Each of the integrity seals applied is composed of a Integrity Seal (IS) field and a Integrity Seal Security Association Identifier (ISSAID) field, as shown in Figure 2.

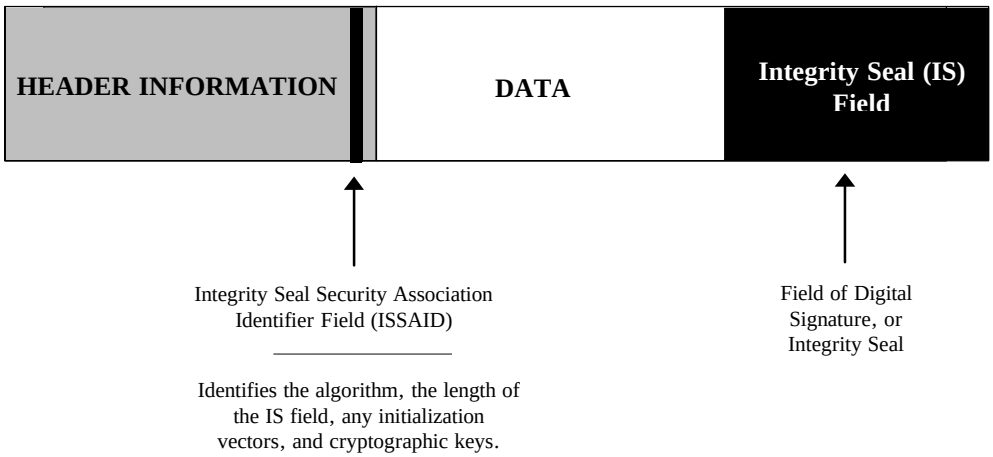

Figure 2. Application of a NITF Integrity Seal to a Header and Data

3.1 THE FILE-LEVEL INTEGRITY SEAL - FILE HEADER BINDING

The NITF file, composed of subheaders, objects, and integrity seals, has a NITF file header that represents it. The NITF file header is strongly bound to the NITF file by the application of an integrity seal to the NITF file header and the rest of the NITF file. The application of the integrity seal to the NITF file header and the rest of the NITF file is optional. The application of a file-level integrity seal is shown in Figure 3.

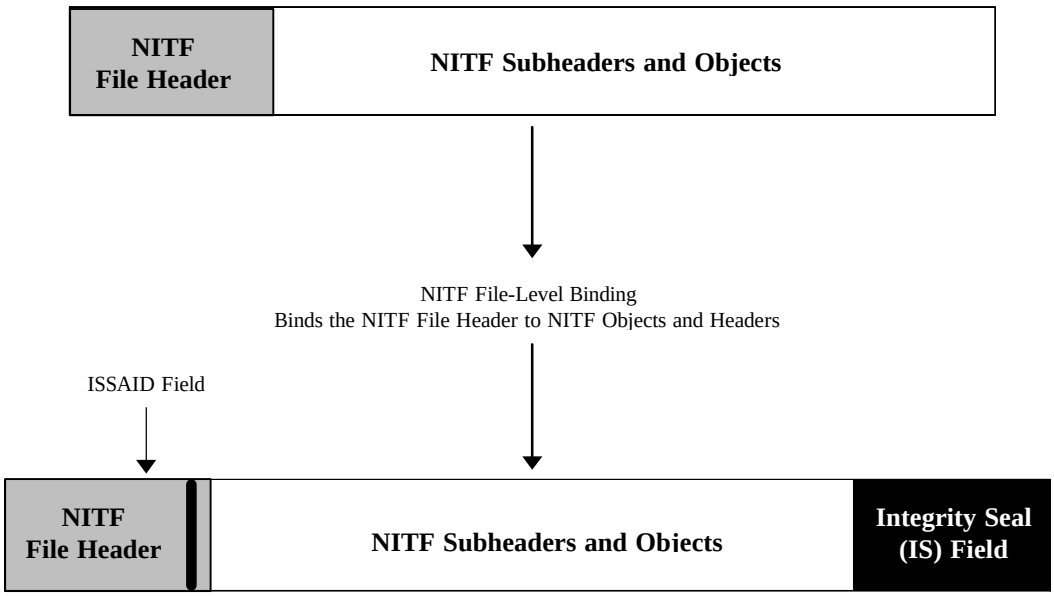

Figure 3. Application of a File-Level Integrity Seal

3.2 OBJECT-LEVEL INTEGRITY SEAL - SUBHEADER BINDING

Each NITF object has an associated subheader. A subheader is strongly bound to the object that it represents by the application of an integrity seal to that subheader and object. The application of integrity seals to subheaders and their respective objects is optional. Figure 4 illustrates the application of object-level integrity seals to a typical NITF file.

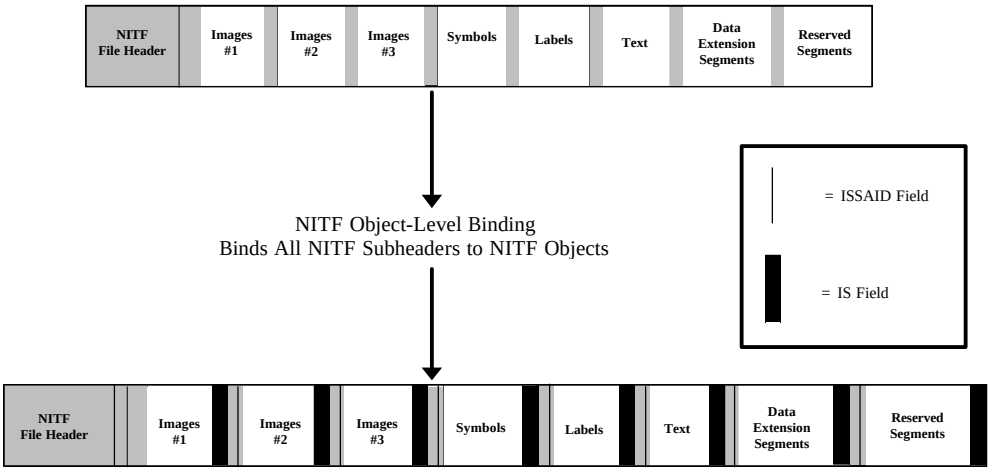

Figure 4. Application of Object-Level Integrity Seals

3.3 SEQUENCE OF BINDINGS

All integrity seals applied to NITF objects must be applied before the application of the NITF file-level integrity seal because the file-level integrity seal is a function of the entire field, including, the object-level integrity seals.

3.4 THE INTEGRITY SEAL SECURITY ASSOCIATION IDENTIFIER

The Integrity Seal Security Association Indicator (ISSAID) is a field in the NITF file header used when a file-level integrity seal is applied and in object subheaders if object-level integrity seals are applied. The purpose of this field is to indicate whether an integrity seal was applied to the NITF header and file or to a subheader and object, the method or algorithm used and its mode, the length of the IS field, the initialization vectors (if any), and the key(s) used with the algorithms to create the seal. The ISSAID field is 32 bytes long and located after the security-relevant fields in each header and subheader. The integrity seal is applied to the NITF file or object with the ISSAID field incorporated within the associated header or subheader. The combination of the ISSAID field and other header information (such as Originator's Name and Originator's Station ID) uniquely identifies a security association. The value of 0 in the header or subheader

indicates that there is no integrity seal applied to the NITF file or the object. Table 1 below shows the addition of the ISSAID field in the headers and subheaders.

FIELD	NAME	SIZE	VALUE RANGE	TYPE
XISSAID	Integrity Seal Security Association Identifier	32	0 = No Integrity Seal 0x0000001-0xffffffff = Security Associations	0

Note: The symbol 'x' in the "FIELD" located above indicates which header is referenced. For example, FISSAID would be contained in the NITF file header, IISSAID would be contained in the image subheader, and TISSAID would be contained in the text subheader.

Table 1. Addition to NITF Headers and Subheader

3.5 THE INTEGRITY SEAL FIELD

The integrity seal consists of data contained in the IS field. The IS field will be appended to the object the seal represents or to the NITF file if it is a file-level integrity seal. The data will be the results of the encrypted one-way hash or keyed one-way hash. The length of this field is based on the algorithm or method used to create the integrity seal. Table 2 is the field specification for the integrity field data.

FIELD	NAME	SIZE	VALUE RANGE	TYPE
xIS	Integrity DataSeal	Variable	0 = No Integrity Seal	0

Note: The symbol 'x' in the "FIELD" located above indicates which header is referenced. For example, FIS would be contained in the NITF file header, IIS would be contained in the image subheader, and TIS would be contained in the text subheader.

Table 2. IS Field Appended to NITF File of Object

4.0 THE DIGITAL SIGNATURE STANDARD

NIST's Digital Signature Standard (DSS) [2] specifies the recommended algorithms and formats applicable to integrity seals. The DSS specifies NIST's Secure Hashing Algorithm (SHA) [3] and NIST's Digital Signature Algorithm (DSA) [2]. The SHA is applied to the concatenation of the header and NITF file of subheader and object. The SHA is signed or encrypted by DSA using a private key to form a digital signature or integrity seal. Verifying the signature or integrity seal requires the header and NITF file or subheader and object, the public key (mathematically related to the private key), and the digital signature. The digital signature will be located in the IS field and will have a length of 80 bytes. The public key will be identified by a security association contained in the ISSAID field.

5.0 ISSUES

The following are among the issues that need to be addressed when implementing the integrity seal for the NITFS.

5.1 KEY MANAGEMENT

The application of integrity seals to NITF files requires cryptographic keys to form the digital signatures. This requires that a key management system be established. This RFC does not propose a specific key management system, however, a key management system is presumed. The association between the key management system and the integrity seal is accomplished by defining security associations to be contained in the ISSAID field.

5.2 OBJECT-LEVEL INTEGRATION SEALS AND FILE-LEVEL INTEGRITY SEAL

The file-level integrity seal binds the NITF file header to the rest of the NITF file. Upon verification of the seal, the verifier can be assured that no headers and subheaders have been altered since the seal was applied. Any authorized modifications to a single object will require that a new fie-level integrity seal be applied, while other object-level integrity seals may remain unmodified.

6.0 REFERENCES

1. MIL-STD-2500, “National Imagery Transmission Format,” Version 2.0, 18 June 1993

2. NIST FIPS PUB 186, “Digital Signature Standard (DSS),” National Institute of Standards and Technology, U.S. Department of Commerce, 19 May 1994.

3. NIST FIPS PUB 180-1, “Secure Hash Standard,” National Institute of Standards and Technology, U.S. Department of Commerce, 31 May 1994.